Embedded indexing: first choice for authors?

**Most indexed technical documents use the technique of embedding.
Bill Johncocks offers his thoughts on its advantages and disadvantages.**

Promising to cover embedded indexing was a little rash as it seems many readers know more about the technique than most professional indexers. Certainly, Mike Unwalla gave an excellent overview of embedding with Microsoft Word in the Autumn 2003 *Communicator* (available from the Members' area of the website). Before we look at Word though, let's examine why embedding should appeal more to technical authors than to most book indexers and publishers.

The indexer's needs
Essentially, it's horses for courses. Book indexers usually start with finished documents—whether as physical page proofs or PDF files—and want a flexible package that can match various publishers' house styles. Where embedding confers no advantage, they'd rather stick with familiar, dedicated standalone indexing packages than learn new tools, especially as theirs usually offer auto-completion and display the growing index as aids to consistency.

The publisher's needs
The time savings achievable through embedding are undeniable though and appeal to many editors in the increasingly hectic world of book publishing. Indexing work-in-progress can ease what's often perceived as a bottleneck near the end of the production cycle, while embedding fits well with a desire to automate the entire production process using XML, so it is making steady if unspectacular inroads into the book world.

The author's needs
For authors, embedding of course has three key advantages that far outweigh any slight loss of flexibility. First, being able to generate an interim index as you work makes even your drafts accessible and helps you check consistency and coverage. Second, embedding protects against last minute changes that affect pagination. Third, your finished index should be at least partially reusable, with updating for new editions simply a matter of indexing the new material and hitting a button.

In practice, though updating is faster, it's seldom completely problem-free. I suspect it's rare for several sections of a document to be replaced with no impact on those retained. Another problem is consistency: even if you're updating a document you indexed yourself, can you recall decisions made months ago? Embedding copes well with minor changes but the effects of major deletions and additions are less predictable. Locators will be faithfully rendered come what may, so the index will still 'work', but the balance of subheadings and unmodified main entries and even, in extreme cases, the choice of main entry terms and the direction of cross-references can alter. Your index fits the document it was compiled for as a glove fits a hand, so the more a document changes, the more thoroughly the new index will need reviewing.

It's also worth pointing out a design limitation of most embedding packages: they automatically supply page numbers as index locators. For documents structured by numbered sections or paragraphs, not only are they unsuitable but the big advantage of embedding—that page numbers are re-assigned with each index-generating run—disappears. A well designed section numbering scheme directs the user more precisely to the desired topic than anything based on page numbers but that's not all: it provides its own flexibility, because the index isn't tied to a specific pagination or layout.

The embedded markup
Embedding uses inbuilt features of word processing, authoring or layout tools to associate index terms with the text they describe, usually by means of hidden text tags. At any stage, the software will build an index, sort entries, suppress duplication and append the appropriate page locators on demand so, as changes to the document change the pagination, re-running the process will always provide the latest page locators.

To help it pick the right page, your insertion point should be near the start of the treatment of a topic. I prefer to put markers after the first word of a sentence, paragraph or picture caption, because some post-processing can strip off exposed embedding, but you might collect all cross-references at the end of a document to avoid their being accidentally lost during text changes.

The final checks
The generated index needs two types of editing: one for consistency and to remove duplication and 'bloat'; another to rationalise subentries and cross-references. Search and replace used on index tag contents is a boon here but requires care. Casual global updating of 'EEC' with 'EU' could cause people to wear *fleues*, make *speuhes* or stand under *beuh* trees as a by-product! Or converting American to British English, might leave not just *metres* in place of *meters* but also *perimeters*, *altimetres*, *thermometres* and parking *metres*. It's safest to include the tag identifier (xε " in Word) in all your search strings but it is still far from straightforward to reverse a *See* cross-reference, change subentries into main entries, or *vice versa*.

It's worth noting symbols and diacritics for a final check because, even after you've edited your index, further processing can cause problems. For example, conversion into QuarkXpress can seriously corrupt Word embedding, as can the attentions of unsympathetic typesetters and page designers!

Sources of help
If you outsource embedding, bear in mind that not all professional indexers are familiar with the technique or with any given software tool. The Society of Indexers' workshop is a one-day event that leaves participants to hone their skills on their own PCs but there is plenty of relevant expertise among ISTC members, while Web searches can supplement the rather variable information in Help files.

In our next issue, we'll look specifically at Word, then return to basics in 2007, because it's proper use of cross-references and subheadings that separates a good index (whether standalone or embedded) from one that is merely serviceable. **C**

Bill Johncocks is a freelance Accredited Indexer living on the Isle of Skye, who specialises in embedded indexing of technical documents.
E: bill.johncocks@btconnect.com
W: www.technicalindexing.com
Society of Indexers: www.indexers.org